

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-250283

(43)Date of publication of application : 17.09.1999

(51)Int.Cl.

G06T 15/70

G06F 3/00

G06T 15/00

// G06F 13/00

(21)Application number : 10-048063

(71)Applicant : INES CORP

(22)Date of filing : 27.02.1998

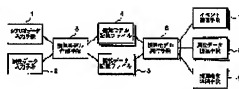
(72)Inventor : YOSHIKAWA SUMUTO

(54) OBJECT EXPRESSING SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a system and a method for expressing a single or plural objects performing in a virtual space.

SOLUTION: This system includes a means 1 for receiving scenario data, a means 2 for receiving attribute data, and a means 3 for preparing a performance model for each object describing its performance. The system further includes a file 4 for storing the performance model, a file 5 for storing attribute data, a means 6 for executing the performance model while transferring data with the file for storing the performance model and the file for storing the attribute data, a means 7 for performing communication of an event linked with the means for executing the performance model, a means 8 for performing communication of the attribute data linked with the means for executing the performance model, and a means 9 for performing communication of a plotting instruction linked with the means for executing the performance model.



特開平11-250283

(43) 公開日 平成11年(1999) 9月17日

(51) Int.Cl.⁵
 G 0 6 T 15/70
 G 0 6 F 3/00
 G 0 6 T 15/00
 // G 0 6 F 13/00

識別記号
 6 5 4
 3 5 1

F I
 G 0 6 F 15/62
 3/00
 13/00
 15/62

3 4 0 K
 6 5 4 A
 3 5 1 E
 3 6 0

審査請求 未請求 請求項の数 6 O L (全 9 頁)

(21) 出願番号 特願平10-48063

(22) 出願日 平成10年(1998) 2月27日

(71) 出願人 595108206

株式会社アイネス
 神奈川県横浜市中区牛久保3-9-2

(72) 発明者 吉川 澄人

神奈川県横浜市中区牛久保3丁目9番2
 号 株式会社アイネス内

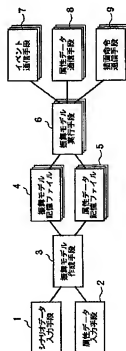
(74) 代理人 弁理士 川口 義雄 (外1名)

(54) 【発明の名称】 オブジェクト表現システム

(57) 【要約】

【課題】 仮想空間内で振る舞う単一または複数のオブジェクトを表現するためのシステムおよび方法を提供する。

【解決手段】 シナリオデータを入力する手段 (1) と属性データを入力する手段 (2) とオブジェクトの振舞を記述する個々のオブジェクトごとの振舞モデルを作成する手段 (3) とを含み、さらに個々のオブジェクトごとに、振舞モデルを記憶するファイル (4) と属性データを記憶するファイル (5) と振舞モデルを記憶するファイルおよび属性データを記憶するファイルとデータをやりとりしながら振舞モデルを実行する手段 (6) と振舞モデルを実行する手段と協働するイベントの通信を行う手段 (7) と振舞モデルを実行する手段と協働する属性データの通信を行う手段 (8) と振舞モデルを実行する手段と協働する描画命令の通信を行う手段 (9) とを含む。



【特許請求の範囲】

【請求項1】 仮想空間内で振る舞う単一または複数のオブジェクトを表現するためのシステムであって、オブジェクトの振舞についてのシナリオデータを入力する手段と、

オブジェクトの振舞についての属性データを入力する手段と、

シナリオデータおよび属性データに基づいて、オブジェクトの振舞を記述する個々のオブジェクトごとの振舞モデルを作成する手段と、

振舞モデルを記憶するファイルと、

属性データを記憶するファイルと、

振舞モデルを記憶するファイルおよび属性データを記憶するファイルとデータをやりとりしながら振舞モデルを実行する手段と、

振舞モデルを実行する手段と協働するイベントの通信を行う手段と、

振舞モデルを実行する手段と協働する属性データの通信を行う手段と、

振舞モデルを実行する手段と協働する描画命令の通信を行う手段とを含み、

振舞モデルを記憶するファイルと属性データを記憶するファイルと振舞モデルを実行する手段とイベントの通信を行う手段と属性データの通信を行う手段と描画命令の通信を行う手段とが個々のオブジェクトごとに設けられているオブジェクトを表現するためのシステム。

【請求項2】 個々のオブジェクトごとの振舞モデルを記憶するファイルにおける個々のオブジェクトごとの振舞モデルが、イベント記述部分と条件記述部分と動作記述部分との一組または複数組を含む請求項1に記載のシステム。

【請求項3】 仮想空間内で振る舞う単一または複数のオブジェクトを表現するための方法であって、オブジェクトの振舞についてのシナリオデータを入力するステップと、

オブジェクトの振舞についての属性データを入力するステップと、

シナリオデータおよび属性データに基づいて、オブジェクトの振舞を記述する個々のオブジェクトごとの振舞モデルを作成するステップと、

振舞モデルを個々のオブジェクトごとにファイルに記憶するステップと、

属性データを個々のオブジェクトごとにファイルに記憶するステップと、

振舞モデルを記憶するファイルおよび属性データを記憶するファイルとデータをやりとりし、イベントの通信を行う手段と属性データの通信を行う手段と描画命令の通信を行う手段と協働しながら個々のオブジェクトごとに振舞モデルを実行するステップとを含むオブジェクトを表現するための方法。

【請求項4】 個々のオブジェクトごとの振舞モデルを記憶するファイルにおける個々のオブジェクトごとの振舞モデルが、イベント記述部分と条件記述部分と動作記述部分との一組または複数組を含む請求項3に記載の方法。

【請求項5】 コンピュータによって仮想空間内で振る舞う単一または複数のオブジェクトを表現するためのプログラムを記録したコンピュータ読み取り可能な記録媒体であって、

コンピュータに、

オブジェクトの振舞についてのシナリオデータを入力するステップと、

オブジェクトの振舞についての属性データを入力するステップと、

シナリオデータおよび属性データに基づいて、オブジェクトの振舞を記述する個々のオブジェクトごとの振舞モデルを作成するステップと、

振舞モデルをファイルに記憶するステップと、

属性データをファイルに記憶するステップと、

振舞モデルを記憶するファイルおよび属性データを記憶するファイルとデータをやりとりし、イベントの通信を行う手段と属性データの通信を行う手段と描画命令の通信を行う手段と協働しながら個々のオブジェクトごとに振舞モデルを実行するステップとを実行させるためのプログラムを記録したコンピュータ読み取り可能な記録媒体。

【請求項6】 個々のオブジェクトごとの振舞モデルを記憶するファイルにおける個々のオブジェクトごとの振舞モデルが、イベント記述部分と条件記述部分と動作記述部分との一組または複数組を含む請求項5に記載のコンピュータ読み取り可能な記録媒体。

【発明の詳細な説明】
【発明の属する技術分野】本発明は、各種シミュレーションやゲームなど幅広い分野で利用される、仮想空間内で振る舞う単一または複数のオブジェクトを表現するためのシステム、方法およびプログラムを記録したコンピュータ読み取り可能な記録媒体に関する。

【従来の技術】仮想空間内のオブジェクトの振舞を記述するには、物理現象、干渉チェック、イベントに基づく処理、アニメーション処理などを記述することが必要である。これらの処理を記述するために、従来は、スクリプトプログラムの記述により行うか、または、別途背景となる空間モデルを用意してそれに対応したパラメータの定義を行う方法が採用されていた。スクリプトプログラムの記述による方法では、オブジェクトの振舞は、並進や回転などの座標変換の関数、あるいは描画用の関数の組合せによって実現される。それらの関数に対して利用者は、計算結果を予測し、適切なパラメータを与えることが必要になる。またこれらの関数を呼び出すタイミングや、条件判断・条件分岐などを、予め連串のプロセ

スとして書き下しておくことによって、アニメーションなどを実現する。空間モデルを使用する方法では、空間上の全てのオブジェクトの状態を管理する機構と、衝突検出や特定の法則に基づく運動を計算する関数群とが、予めデータベースなどに用意されている。利用者はオブジェクトと利用する関数との組合せ、および必要なパラメータをこれらに対して登録する。例えばオブジェクトの移動経路（パス）を表す曲線などを、パラメータにより制御する。

【発明が解決しようとする課題】しかしながら、従来技術のスクリプトプログラムでの記述による方法では、数学的知識とプログラミング技術が必要であるため、一般の利用者の負荷が大きくなる。また、予めプログラムを設定しておくため、利用者がシステム稼働中に振舞を動的に変更することができない。一方、空間モデルを使用する方法では、空間モデルの制約を受けるため、振舞表現の多様性やインタラクションの柔軟性に乏しい。また、空間モデルのサポートされる環境でしか稼働させることができない。本発明は、このような問題点を解決し、一般の利用者が容易に、しかも特別な知識を必要とせずに、システム稼働中であっても、振舞表現の多様性やインタラクションの柔軟性を確保しつつ、仮想空間内のオブジェクトの振舞を記述することができるシステムを提供することを目的とする。

【課題を解決するための手段】本発明のシステムは、オブジェクトの振舞についてのシナリオデータを入力する手段と、オブジェクトの振舞についての属性データを入力する手段と、シナリオデータおよび属性データに基づいて、オブジェクトの振舞を記述する個々のオブジェクトごとの振舞モデルを作成する手段と、振舞モデルを記憶するファイルと、属性データを記憶するファイルと、振舞モデルを記憶するファイルおよび属性データを記憶するファイルとデータとをやりとりしながら振舞モデルを実行する手段と、振舞モデルを実行する手段と協調するイベントの通信を行う手段と、振舞モデルを実行する手段と協調する属性データの通信を行う手段と、振舞モデルを実行する手段と協調する描画命令の通信を行う手段とを含む、振舞モデルを記憶するファイルと属性データを記憶するファイルと振舞モデルを実行する手段とイベントの通信を行う手段と属性データの通信を行う手段と描画命令の通信を行う手段とが個々のオブジェクトごとに設けられていることを特徴とする。また、本発明の方法は、オブジェクトの振舞についてのシナリオデータを入力するステップと、オブジェクトの振舞についての属性データを入力するステップと、シナリオデータおよび属性データに基づいて、オブジェクトの振舞を記述する個々のオブジェクトごとの振舞モデルを作成するステップと、振舞モデルを個々のオブジェクトごとにファイルに記憶するステップと、属性データを個々のオブジェクトごとにファイルに記憶するステップと、振舞モデルを

記憶するファイルおよび属性データを記憶するファイルとデータをやりとりし、イベントの通信を行う手段と属性データの通信を行う手段と描画命令の通信を行う手段と協調しながら個々のオブジェクトごとに振舞モデルを実行するステップとを含むことを特徴とする。したがって、本発明によれば、利用者は、シナリオデータを入力することによって容易にオブジェクトの振舞を定義することができる。また、本発明によれば、個々のオブジェクトごとの振舞モデルを作成するので、物理現象にしたがう振舞から非現実的な振舞まで、多様な振舞を表現することができる。さらに、本発明によれば、個々のオブジェクトごとの振舞モデルを作成しながら実行するので、利用者は、システム稼働中に、シナリオデータを入力することによって振舞モデルを動的に変更することができる。さらに、本発明によれば、振舞モデルを実行する手段が、属性データ通信手段を介して、属性データを相互に変更することによってオブジェクト間の相互作用が表現されるので、オブジェクト間のインタラクション（相互作用）の表現の柔軟性を確保するとともに、システムの構造および方法の手順が簡単となる。本発明のシステムおよび方法において、好ましくは、個々のオブジェクトごとの振舞モデルを記憶するファイルにおける個々のオブジェクトごとの振舞モデルが、イベント記述部分と条件記述部分と動作記述部分との一組または複数組を含む。したがって、本発明によれば、オブジェクト間の柔軟なインタラクション（相互作用）を簡単に表現することができる。本発明は、限定的ではなく例示的に示した、添付の図面を参照して説明する好ましい実施形態に関する下記の記載によりさらによく理解することができる。

【発明の実施の形態】図1に本発明のシステムの構成例を示す。本発明のシステムは、シナリオデータ入力手段1と、属性データ入力手段2と、振舞モデル作成手段3と、振舞モデル記憶ファイル4と、属性データ記憶ファイル5と、振舞モデル実行手段6と、イベント通信手段7と、属性データ通信手段8と、描画命令通知手段9とを含む。まず、振舞モデルの作成手順について図1のシステム構成図と図3のフローチャートとを参照しながら説明する。利用者は、シナリオデータ入力手段1を通じてシナリオデータを入力する（図3のステップ304）。シナリオの具体例については、後で説明する。振舞モデル作成手段3は、入力されたシナリオデータを、個々のオブジェクトの振舞モデルとして定義し直し（図3のステップ306）、個々のオブジェクトに割り振る（図3のステップ307）。具体的に、たとえば、「物体Aが物体Bを運ぶ」というシナリオにおける相互作用は、「物体Aは（利用者のマウスドラッグなどで）移動する」という物体Aを主体とする振舞モデルと「物体Bは物体Aと接触したとき物体Aと同様に移動する」という物体Bを主体とする振舞モデルとに定義し直された後、

物体Aと物体Bという個々のオブジェクトに割り振られる。個々のオブジェクトごとの振舞モデル記憶ファイル4が振舞モデルを記憶する。利用者は、属性データ入力手段2を通じて属性データを入力する。あるいは属性データ入力手段2は、別途作成されたコンピュータグラフィックスのデータから属性データを取得することできる。振舞モデル作成手段3は、入力された属性データをオブジェクト毎に、個々のオブジェクトごとの属性データとして割り振る(図3のステップS307)。個々のオブジェクトごとの属性データ記憶ファイル5が属性データを記憶する。ここで、図1から明らかなように、振舞モデル記憶ファイル4と、属性データ記憶ファイル5と、振舞モデル実行手段6と、イベント通信手段7と、属性データ通信手段8と、描画命令通知手段9とは、個々のオブジェクトごとに設けられている。図2は、振舞モデル記憶ファイル4と振舞モデル実行手段6との詳細を示す。振舞モデルは、いわゆるECAルールに基づき、イベント記述部(Event)401と条件記述部(Condition)402と動作記述部(Action)403とから構成される。なお、ECAルールについては、「発明の実施の形態」の最後で説明する。Eventには、イベントを発行したオブジェクト(Who)およびイベントの種別(What)を記述する。Conditionには、条件判断の対象となるオブジェクト(Who)、属性データ変化量計算関数群から起動するメソッドの種別(What)および属性データが満たすべき条件(How)を記述する。Actionには、動作実行の主体となるオブジェクト(Who)、属性データ変化量計算関数群から起動するメソッドの種別(What)および属性データが満たすべき条件(How)を記述する。図2に示した、振舞モデル記憶ファイル4と、属性データ記憶ファイル5と、振舞モデル実行手段6と、イベント通信手段7と、属性データ通信手段8と、描画命令通知手段9と、属性データ変化量計算関数群10とは、個々のオブジェクトごとに設けられている。また、個々のオブジェクトごとの振舞モデル記憶ファイル4の中には、ECAルールによって表現された振舞モデルの、イベント記述部と条件記述部とのWhoは、必ずしも、オブジェクトCに一致しないが、動作記述部のWhoは、必ず、オブジェクトCに一致する。すなわち、動作記述部のWhoが、個々のオブジェクトを表す。上記の「物体Bは物体Aと接触したとき物体Aと同様に移動する」という物体Bを主体とする振舞モデルにおいて、第1に、物体Aまたは物体Bの位置変化がイベント記述部によって表現される。第2に、物体Aと物体Bとの接触が条件記述部によって表現される。そして、第3に、物体Bの物体Aと同様の移動が動作記述部によって表現される。動作記述部のWhoは、物体Bである。条件記述部や、動作記述部における使用するメソッ

ドは、属性データ変化量計算関数群と呼ぶファンクションを使用して定義する。ここで、属性データ変化量計算関数群とは、オブジェクトの属性データの変化量を計算するための関数群である。たとえば、位置や姿勢の変化、物理法則に基づく変化、衝突検出などの計算を行う。各関数はオブジェクト指向に基づいて設計され、クラスオーバーメソッドのままとりとして記憶されている。また、ここで属性データとは、オブジェクトの描画に用いられる属性とその他の物理特性などを表す属性である。前者は、たとえばオブジェクトの形状、位置、姿勢、スケール、色、反射光、テクスチャ、構成部品、イベントセンサなどである。これらは3次元コンピュータグラフィックスの分野において、常套的に用いられる属性である。後者はたとえば、オブジェクトの名称、質量、材質、初速度などである。つぎに、図2のシステム構成図および図4のフローチャートを参照しながら、振舞モデルの実行手順について説明する。振舞モデル実行手段6のイベント照合部601は、イベント通信手段7よりイベントの通知を受け取る(図4のステップS401)。イベントは、後に説明するように、他のオブジェクトの実行手段または、利用者のトリガによって発生する。イベント照合部601は、通知されたイベントと振舞モデル記憶ファイル4の振舞モデルのイベント記述部401との照合を行い(図4のステップS403)、一致したイベントを持つ振舞モデルを呼出す。一致したイベントを持つ振舞モデルがあれば以下の処理を行わない。振舞モデル実行手段6の条件判断部602は、イベントが一致した振舞モデルの条件記述部402を呼び出す(図4のステップS404)。振舞モデル実行手段6の条件判断部602は、当該オブジェクトモデルの属性データを受け取る(図4のステップS405)、属性データ通信手段8を通じて他のオブジェクトの必要な属性データを受け取る(図4のステップS407)。振舞モデル実行手段6の条件判断部602は、得られた属性データをパラメータとして属性データ変化量計算関数群10のメソッドを起動する(図4のステップS408)。属性データ変化量計算関数群10は、属性データの変化量を計算し、計算結果を振舞モデル実行手段6の条件判断部602に返す。振舞モデル実行手段6の条件判断部602は、得られた計算結果と振舞モデルの条件記述部402との照合を行う(図4のステップS409)。条件が一致しなければ以下の処理を行わない。振舞モデル実行手段6の動作実行部603は、振舞モデルの動作記述部403を呼び出す(図4のステップS410)。振舞モデル実行手段6の動作実行部603は、当該オブジェクトモデルの属性データを受け出す(図4のステップS411)、属性データ通信手段8を通じて他のオブジェクトの必要な属性データを受け取る(図4のステップS413)。振舞モデル実行手段6の動作実行部603は、得られた属性デー

タをパラメータとして属性データ変化量計算関数群のメソッドを起動する（図4のステップS414）。属性データ変化量計算関数群10は、属性データの変化量を計算し、計算結果を振舞モデル実行手段6の動作実行部603に返す。振舞モデル実行手段6の動作実行部603は、計算結果をもとに属性データの更新を行う（図4のステップS415）。振舞モデル実行手段6の動作実行部603は、イベント通信手段7を通じて、他オブジェクトに属性データの更新があったことを通知する（図4のステップS416）。振舞モデル実行手段6の動作実行部603は、計算結果をもとに描画命令通知手段9を起動する（図4のステップS417）。描画命令通知手段9は、描画命令を発行する。描画は、3次元コンピュータグラフィックスの描画機能に、更新された属性データを受け渡すことで行う。つぎに、本発明のオブジェクト表現システムをゲームに適用した具体例を以下に説明する。この具体例においては、描画にVRMLを使用している。なお、VRMLについては、「発明の実施の形態」の最後で説明する。ゲームの画面構成の一例を図5に示す。ゲームの内容は、以下のとおりである。ゲームは、周囲を壁に囲まれた四角いフィールド内11で行われる。4面の壁12のうち対向する2面の一部にゴール13が設けられている。2名のプレーヤ14が相手方のゴールへボールを入れるようにフィールド内を移動する。プレーヤ14は、ゲームを行う者の操作により移動する。ボール15は、プレーヤ14に接触すると反発して移動する。また、フィールド11内の数枚には、ランダムに移動する障害16が設けられている。障害は、プレーヤに近づくと接触しないように方向を変える。ボール15は、障害16に接触した場合は反発して移動する。ボール15がどのように反発するかは、ボール15自身の移動速度と移動方向およびボール15と接触したプレーヤ14または障害16の移動速度と移動方向にしたがって変化する。また、プレーヤは、フィールド内において、互いのゴール近辺の限られた範囲を移動する。したがって、プレーヤ同士が接触することはない。さらに、アイテム17が常時ランダムにフィールド上を移動している。ここで、プレーヤ14がアイテム17と接触すると、ボール15は10秒間黒色になり、この間ボール15は障害16と相手方のプレーヤ14とを通り抜けることができるというルールを設ける場合を考える。この振舞は本発明のシステムによって記述する。まず、シナリオデータ的具体例を図6に示す。図6から明らかなように、シナリオデータは、自然言語に近いユーザにとってきわめて理解しやすいものである。利用者は、このようなシナリオデータをシナリオデータ入力手段1から入力する。その後、振舞モデル作成手段3が、シナリオデータおよび属性データに基づいて、個々のオブジェクトごとの振舞モデルを作成する。図6のシナリオデータに対応する、個々のオブジェクトごとの振舞の記述例を図7

に示す。この場合、個々のオブジェクトはボールである。図7において、第1のECAルールは、プレーヤとアイテムとの位置が変わった場合に、プレーヤとアイテムとが衝突したならば、ボールのIDフラグをオンすることを記述する。第2のECAルールは、ボールのIDフラグが変わった場合に、IDフラグがオンであったならば、10秒間計時することを記述する。第3のECAルールは、ボールのIDフラグがオンとなった10秒間経過した場合に、IDフラグがオンであったならば、IDフラグをオフとすることを記述する。第4のECAルールは、ボールのIDフラグが変わった場合に、IDフラグがオンであったならば、ボールの色を黒色に変えることを記述する。第5のECAルールは、ボールのIDフラグが変わった場合に、IDフラグがオンであったならば、ボールが障害と相手方のプレーヤとを通り抜けることができることを記述する。第1乃至第5のECAルールにおいて、動作記述部のWhoは、すべてボールである。このように、振舞モデルは、ECAルールに基づいて、オブジェクトごとに記述されているが、属性データの授受が属性データ通信手段を介して、オブジェクト間で行われることによって、複数のオブジェクトの間の相互作用が表現される。ここで、本明細書で使用した用語について説明する。ECAルールは、いわゆるエキスパートシステムを稼働させるプロダクション・ルールの記述方法として提唱された。動作の起動から実行までの手順を、イベント検出(Event)―条件判断(Condition)―Action(動作実行)のセットにより表現する。ECAルールを用い、データベースを自動的に、連続的に更新するといった応用が研究されている。VRML (Virtual Reality Modeling Language) は、WWW (World Wide Web) 上で3次元コンピュータ・グラフィックスを表示するための言語で、HTML (Hyper Text Markup Language) 文書と同様に、ハイパーテキストの形で利用する。各国のベンダー企業と大学で構成される団体VAG (VRML Architecture Group) において、国際標準化が推進されている。

【発明の効果】本発明によれば、利用者は、シナリオデータを入力することによって容易にオブジェクトの振舞を定義することができる。したがって、たとえば、本発明のシステムまたは方法をゲームに適用した場合には、プログラムの知識・経験のない利用者でも、考案したゲームのアイデアを容易にシステム上に再現することが可能になる。また、物理現象などのシミュレーションに適用した場合には、プログラムの知識・経験のない利用者でも、容易にシナリオデータのパラメータを設定し、オブジェクトの振舞を制御することが可能になる。さらに、プレゼンテーションに適用した場合、プログラムの知識・経験のない利用者でも、たとえばWWW上で流通させるコンテンツや、映画やコーポレート・フィルムにおける演出の視覚効果を検討する際に、登場させるオブジェクトの振舞を容易に設計することが可能になる。ま

た、本発明によれば、個々のオブジェクトごとの振舞モデルを作成するので、物理現象にしたがう振舞から非現実的な振舞まで、多様な振舞を表現することができる。したがって、たとえば、本発明のシステムまたは方法はゲームに適用した場合には、ルールの条件に応じて特定のオブジェクトの振舞を変化させたり、ゲームのレベルに合わせて特定のオブジェクトに複雑な振舞をさせることが可能になる。また、物理現象などのシミュレーションに適用した場合には、たとえば、同一の空間上に登場する、ある条件や法則にしたがうオブジェクトとしたがわいオブジェクトとで、振舞を比較するようなシミュレートや、特定のオブジェクトに複数の振舞を行わせたり、複数のオブジェクトに同一の振舞を行わせるシミュレーションを行うことが可能になる。さらに、プレゼンテーションに適用した場合、特定のオブジェクトの振舞をカリカチュア的に強調して表現することが可能になる。さらに、本発明によれば、個々のオブジェクトごとの振舞モデルを作成しながら実行するので、利用者は、システム稼働中に、振舞モデルを動的に変更することができる。したがって、たとえば、本発明のシステムまたは方法をゲームに適用した場合には、利用者がシナリオデータを変更し、自らゲームのルールを変更することが可能になる。また、物理現象などのシミュレーションに適用した場合には、利用者がシナリオデータを変更し、オブジェクトの性質を変更しながらシミュレーションを繰り返すことができる。さらに、プレゼンテーションに適用した場合、利用者がシナリオデータを変更し、聴衆の要求に応じて場面設定を変更することが可能になる。さらに、本発明によれば、振舞モデルを実行する手段が、属性データの変更を行うことによってオブジェクト間の相互作用が表現されるので、オブジェクト間のインタラクション（相互作用）の表現の柔軟性を確保するとともに、システムの構造および方法の手順が簡単となる。したがって、本発明のシステムまたは方法をゲームや物理現象などのシミュレーション、あるいはプレゼンテーションに適用した場合には、利用者がオブジェクトの追加や抹消などを行い、シナリオデータを変更させる際に、空間全体の構造に依らず、オブジェクト間の関係のみが変更されるため、空間の制約を受けることなく容易にオブジェクト間の相互作用を実現することができる。さらに、本発明によれば、好ましくは、振舞モデルがイベント記述部分と、条件記述部分と、動作記述部分とによ

てオブジェクトの振舞を記述するので、オブジェクト間の柔軟なインタラクション（相互作用）を簡単に表現することができる。したがって、たとえば、本発明のシステムまたは方法をゲームに適用した場合には、利用者が入力するシナリオデータを、イベント記述、条件記述、動作記述の組合せによって再定義するため、複雑なルールにしたがうオブジェクトの振舞やオブジェクト間の相互作用を容易に表現することが可能になる。また、物理現象などのシミュレーションに適用した場合には、利用者が入力するシナリオデータを、イベント記述、条件記述、動作記述の組合せによって定義し直すため、複雑な条件や法則にしたがうオブジェクトの振舞やオブジェクト間の相互作用を容易に表現することが可能になる。さらに、プレゼンテーションに適用した場合、利用者が入力するシナリオデータを、イベント記述、条件記述、動作記述の組合せによって定義し直すため、複合的な動きを伴うアニメーションなどにおける、オブジェクトの振舞やオブジェクト間の相互作用を容易に表現することが可能になる。

【図面の簡単な説明】

【図1】本発明のシステム全体の構成を示す図である。

【図2】本発明のシステムの振舞モデル実行手段および振舞モデル記憶ファイルの構成を示す図である。

【図3】本発明の方法の振舞モデルの作成手順を示す図である。

【図4】本発明の方法の振舞モデルの実行手順を示す図である。

【図5】ゲーム画面の構成の一例を示す図である。

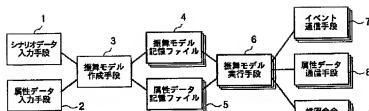
【図6】本発明で使用するシナリオデータの具体例を示す図である。

【図7】本発明における個々のオブジェクトの振舞の記述例を示す図である。

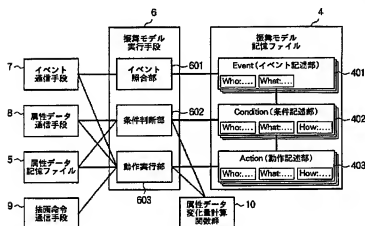
【符号の説明】

- 1 シナリオデータ入力手段
- 2 属性データ入力手段
- 3 振舞モデル作成手段
- 4 振舞モデル記憶ファイル
- 5 属性データ記憶ファイル
- 6 振舞モデル実行手段
- 7 イベント通信手段
- 8 属性データ通信手段
- 9 描画命令通知手段

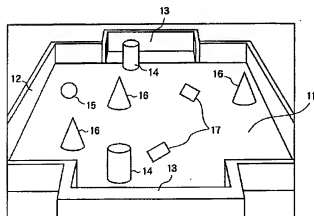
【図 1】



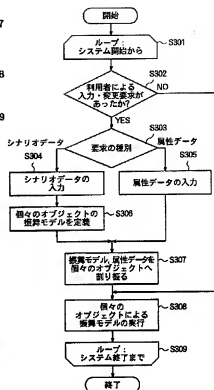
【図 2】



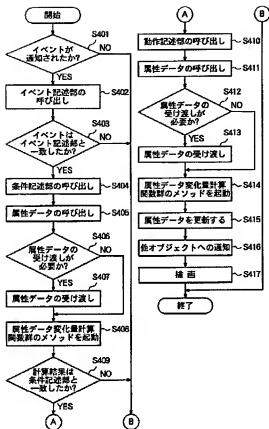
【図 5】



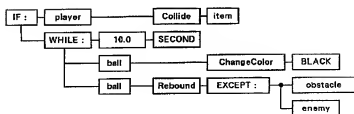
【図 3】



【图4】



【图6】



【図 7】

```

EVENT : : WHO : player/item ; WHAT : Position ;
CONDITION : : WHO : player/item ; WHAT : Position/checkCollision ; HOW : Collided ;
ACTION : : WHO : ball ; WHAT : Flag/setFlag ; HOW : ID/ON ;

EVENT : : WHO : ball ; WHAT : Flag ;
CONDITION : : WHO : ball ; WHAT : Flag/checkFlag ; HOW : ID/ON ;
ACTION : : WHO : ball ; WHAT : Time/countInterval ; HOW : 10.0/SECOND ;

EVENT : : WHO : ball ; WHAT : Interval ;
CONDITION : : WHO : ball ; WHAT : Flag/checkFlag ; HOW : ID/ON ;
ACTION : : WHO : ball ; WHAT : Flag/setFlag ; HOW : ID/OFF ;

EVENT : : WHO : ball ; WHAT : Flag ;
CONDITION : : WHO : ball ; WHAT : Flag/checkFlag ; HOW : ID/ON ;
ACTION : : WHO : ball ; WHAT : Color/changeColor ; HOW : BLACK ;

EVENT : : WHO : ball ; WHAT : Flag ;
CONDITION : : WHO : ball ; WHAT : Flag/checkFlag ; HOW : ID/ON ;
ACTION : : WHO : ball ; WHAT : Direction/rebound ; HOW : EXCEPT/obstacle/enemy ;

```